# *Supplementary Materials*: Person Search by Text Attribute Query as Zero-Shot Learning

Qi Dong
Queen Mary University of London
q.dong@qmul.ac.uk

Shaogang Gong
Queen Mary University of London
s.gong@qmul.ac.uk

Xiatian Zhu
Vision Semantics Ltd.
eddy.zhuxt@gmail.com

As described in the paper, the proposed *Attribute-Image Hierarchical Matching* (AIHM) model consists of 4 components: (1) Hierarchical visual embedding (Sec. 1), (2) Hierarchical textual embedding (Sec. 2), (3) Cross-modality cross-level embedding (Sec. 3), and (4) Matching module (Sec. 4). We detail the network design of all the components below. The notations are consistent with the main paper. For facilitating the presentation, we start by giving the embedding dimensions as summarised in Table 1.

Table 1: Embedding dimensions.

| Definition | Notation | Value |
|---|---|---|
| Local embedding dimension | $\text{Dim}_{emb}^{\text{loc}}$ | 512 |
| Global embedding dimension | $\text{Dim}_{emb}^{\text{glo}}$ | 1024 |
| Cross-modal embedding dimension | $\text{Dim}_{emb}^{\text{s}}$ | 512 |

## 1. Hypercritical Visual Embedding Network

We describe the details of the 2-layers and 4-layers hierarchical visual embedding.

### 1.1. 2-Layers Hierarchical Visual Embedding

In the main experiments, we employ a 2-layers multi-task learning design for hierarchical visual embedding. The architecture details are shown in Figure 1 with the layer configurations listed in Table 2.

Table 2: Configuration of 2-layers visual embedding.

| Structure | Size |
|---|---|
| ResNet-50 | Output size is 2048 |
| $\text{FC}_1$ | 2048 $\times\text{Dim}_{emb}^{\text{glo}}$<br>Tanh |
| $\text{FC}_{1,i}$ (i=1, 2, ..., $N_{\text{Attr}}$) | 2048 × 1024<br>ReLU |
| $\text{FC}_{2,i}$ (i=1, 2, ..., $N_{\text{Attr}}$) | 1024 × $\text{Dim}_{emb}^{\text{loc}}$ |
| $\text{Classification}_i$ (i=1, 2, ..., $N_{\text{Attr}}$) | $\text{Dim}_{emb}^{\text{loc}} \times N_{\text{Attr}_i}$ |

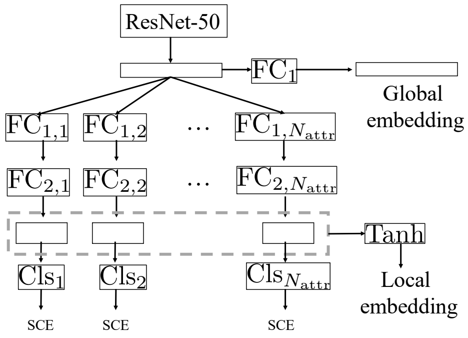


Figure 1: Design of 2-layers visual embedding. Cls: Classification. SCE: Softmax Cross Entropy loss function.

### 1.2. 4-Layers Hierarchical Visual Embedding

The 4-layers hierarchical visual embedding is by a tree-structured multi-task learning design. The architecture design is shown in Figure 2 with the layer configuration listed in Table 3.

Table 3: Configuration of 4-layers visual embedding.

| Structure | Size |
|---|---|
| ResNet-50 | Output size is 2048 |
| $\text{FC}_1$ | 2048 × $\text{Dim}_{emb}^{\text{glo}}$<br>Tanh |
| $\text{FC}_{1,i}$ (i=1, 2) | 2048 × 1024<br>ReLU |
| $\text{FC}_{2,i}$ (i=1, 2, 3, 4) | 1024 × 512<br>ReLU |
| $\text{FC}_{3,i}$ (i=1, 2, ..., $N_{\text{Attr}}$) | 512 × $\text{Dim}_{emb}^{\text{loc}}$ |
| $\text{Classification}_i$ (i=1, 2, ..., $N_{\text{Attr}}$) | $\text{Dim}_{emb}^{\text{loc}} \times N_{\text{Attr}_i}$ |

## 2. Hypercritical Textual Embedding Network

The textual embedding consists of two parts: (1) local textual embedding and (2) global textual embedding. Simi-

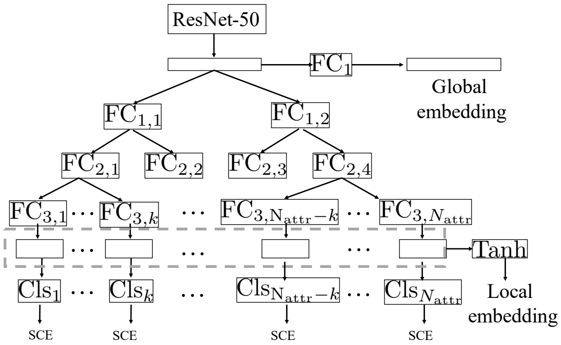

ResNet-50
$FC_1$
Global embedding
$FC_{1,1}$
$FC_{1,2}$
$FC_{2,1}$
$FC_{2,2}$
$FC_{2,3}$
$FC_{2,4}$
$FC_{3,1}$ ··· $FC_{3,k}$ ··· $FC_{3,N_{attr}-k}$ ··· $FC_{3,N_{attr}}$
Tanh
$Cls_1$ ··· $Cls_k$ ··· $Cls_{N_{attr}-k}$ ··· $Cls_{N_{attr}}$
Local embedding
SCE
SCE
SCE
SCE

Figure 2: Design of 4-layers visual embedding. Cls: Classification. SCE: Softmax Cross Entropy loss function. The hierarchy is built as a balanced binary tree structure as possible, i.e. k $\simeq N_{\text{attr}}/4$.

larly, we describe 2-layers and 4-layers hierarchical textual embedding, respectively.

### 2.1. 2-Layers Hierarchical Textual Embedding

In textual embedding, the input is a set of text attributes. Each text attribute is firstly passed into a word2vector model trained on wikipedia [1] and then into three FC layers, The resulting local embeddings are then utilised to form the global embedding. See the architecture in Figure 3 and layer configuration in Table 4.

Table 4: Configuration of 2-layers textual embedding. The setting of Conv layers: the number of input channels, the number of output channels, kernel size, stride, and padding. Cls: Classification.

| Structure | Size |
|---|---|
| $FC_1$ | $300 \times 512$<br>Tanh |
| $FC_2$ | $512 \times 1024$<br>Tanh |
| $FC_3$ | $1024 \times \text{Dim}^{\text{loc}}_{emb}$<br>Tanh |
| $\text{Cls}_i$ (i=1, 2, ..., $N_{\text{Attr}}$) | $\text{Dim}^{\text{loc}}_{emb} \times N_{\text{Attr}_i}$ |
| $\text{Fusion}_1$ | Conv[$\text{Dim}^{\text{loc}}_{emb}$, $\text{Dim}^{\text{glo}}_{emb}$, 1, 1, 0]<br>Conv[$N_{\text{Attr}}$, 1, 1, 1, 0]<br>Tanh |

### 2.2. 4-Layers Hierarchical Textual Embedding

The 4-layers textual embedding is in a similar structure as the 2-layers counterpart. See the architecture in Figure 4 and layer configuration in Table 5.

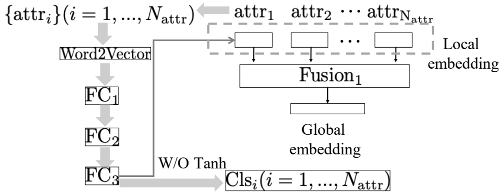

$\{\text{attr}_i\}(i = 1, ..., N_{\text{attr}})$
$\text{attr}_1$ $\text{attr}_2$ ··· $\text{attr}_{N_{\text{attr}}}$
Word2Vector
Local embedding
$\text{Fusion}_1$
$FC_1$
$FC_2$
Global embedding
W/O Tanh
$FC_3$
$\text{Cls}_i(i = 1, ..., N_{\text{attr}})$

Figure 3: Design of 2-layers textual embedding. Cls: Classification.

Table 5: Configuration of 4-layers textual embedding. Cls: Classification.

| Structure | Size |
|---|---|
| $FC_1$ | $300 \times 512$<br>Tanh |
| $FC_2$ | $512 \times 1024$<br>Tanh |
| $FC_3$ | $1024 \times \text{Dim}^{\text{loc}}_{emb}$<br>Tanh |
| $\text{Cls}_i$ (i=1, 2, ..., $N_{\text{Attr}}$) | $\text{Dim}^{\text{loc}}_{emb} \times N_{\text{Attr}_i}$ |
| $\text{Fusion}_{1,i}$ (i=1,2,3,4) | Conv[$\text{Dim}^{\text{loc}}_{emb}$, 512, 1, 1, 0]<br>Conv[k,1,1,1,0]<br>Tanh |
| $\text{Fusion}_{2,i}$ (i=1,2) | Conv[512, 512, 1, 1, 0]<br>Conv[2,1,1,1,0]<br>Tanh |
| $\text{Fusion}_3$ | Conv[512, $\text{Dim}^{\text{glo}}_{emb}$, 1, 1, 0]<br>Conv[2,1,1,1,0]<br>Tanh |

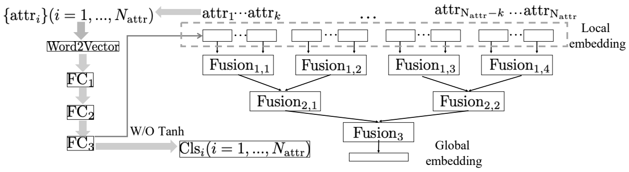


Figure 4: Design of 4-layers textual embedding. Cls: Classification. The hierarchy is built as a balanced binary tree structure as visual embedding, i.e. k $\simeq N_{\text{attr}}/4$.

## 3. Cross-Modality Cross-Level Embedding

Given the hierarchical visual and textual embedding, we then conduct global-level and local-level cross-modality embedding followed with cross-level cross-modality embedding. The configuration of layers are listed in Table 6.

### 3.1. Cross-Modality Global-Level Embedding

The global-level fusion module takes as input the global visual embedding $\boldsymbol{x}^{\text{glo}}$ and the global textual embedding

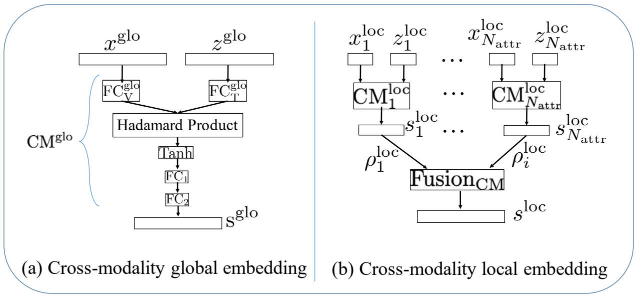
Table 6: Configuration of cross-level (CL) cross-modality (CM) embedding.

| Structure | Size |
|---|---|
| $\mathrm{FC}^{\mathrm{i}}_{\mathrm{T/V}}$, $i \in$ {glo,loc} | $\mathrm{Dim}^{j}_{emb} \times 512$ |
| $\mathrm{FC}_1$ | $512 \times 512$<br>Tanh |
| $\mathrm{FC}_2$ | $512 \times 512$<br>Tanh |
| $\mathrm{Fusion}_{\mathrm{CM}}$ | Conv[512,$\mathrm{Dim}^{\mathrm{s}}_{emb}$, 1, 1, 0]<br>Conv[$N_{\mathrm{attr}}$, 1, 1, 1, 0]<br>Tanh |
| $\mathrm{Fusion}_{\mathrm{CL}}$ | Conv[512, $\mathrm{Dim}^{\mathrm{s}}_{emb}$, 1, 1, 0]<br>Conv[2, 1, 1, 1, 0]<br>Tanh |

$\boldsymbol{z}^{\mathrm{glo}}$, outputting the global cross-modality embedding $\boldsymbol{s}^{\mathrm{glo}}$. The architecture is shown in Figure 5 (a).

### 3.2. Cross-Modality Local-level Embedding

The local-level fusion module takes as input the local visual embedding $\{\boldsymbol{x}_i^{\mathrm{loc}}\}_{i=1}^{N_{\mathrm{att}}}$ and the local textual embedding $\{\boldsymbol{z}_i^{\mathrm{loc}}\}_{i=1}^{N_{\mathrm{att}}}$, outputting the local cross-modality embedding $s^{\mathrm{loc}}$. The architecture is given in Figure 5 (b).


$x^{\mathrm{glo}}$
$z^{\mathrm{glo}}$
$\mathrm{FC}^{\mathrm{glo}}_{\mathrm{V}}$
$\mathrm{FC}^{\mathrm{glo}}_{\mathrm{T}}$
Hadamard Product
$\mathrm{CM}^{\mathrm{glo}}$
Tanh
$\mathrm{FC}_1$
$\mathrm{FC}_2$
$\mathrm{s}^{\mathrm{glo}}$
$x_1^{\mathrm{loc}}$
$z_1^{\mathrm{loc}}$
$x_{N_{\mathrm{attr}}}^{\mathrm{loc}}$
$z_{N_{\mathrm{attr}}}^{\mathrm{loc}}$
$\mathrm{CM}_1^{\mathrm{loc}}$
$\mathrm{CM}_{N_{\mathrm{attr}}}^{\mathrm{loc}}$
$s_1^{\mathrm{loc}}$
$s_{N_{\mathrm{attr}}}^{\mathrm{loc}}$
$\rho_1^{\mathrm{loc}}$
$\rho_i^{\mathrm{loc}}$
$\mathrm{Fusion}_{\mathrm{CM}}$
$s^{\mathrm{loc}}$

(a) Cross-modality global embedding

(b) Cross-modality local embedding

Figure 5: Design of cross-modality (a) global-level and (b) local-level embedding.

### 3.3. Cross-Modality Cross-Level Embedding

Given the global $s^{\mathrm{glo}}$ and local $s^{\mathrm{loc}}$ cross-modality embedding, we obtain the cross-modality cross-level embedding $\boldsymbol{s}$ as shown in Figure 6.

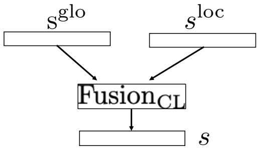


Figure 6: Design of cross-level (CL) embedding.

## 4. Matching Module

The matching module takes as input the cross-modality cross-level embedding $\boldsymbol{s}$, and outputs the similarity score $\hat{y} \in [0, 1]$ of the input image and attribute set. In training, we set the ground-truth similarity score 1 for the *matching* attribute-image pairs and 0 for the *unmatched* attribute-image pairs. The details are shown in Table 7 and Figure 7.

Table 7: Configuration of matching module.

| Structure | Size |
|---|---|
| $\mathrm{FC}_1$ | $\mathrm{Dim}^{\mathrm{s}}_{emb} \times 256$<br>ReLU |
| $\mathrm{FC}_2$ | $256 \times 128$<br>ReLU |
| $\mathrm{FC}_3$ | $128 \times 1$<br>Sigmoid |

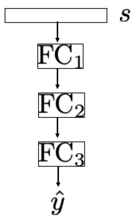


Figure 7: Design of matching module.

## References

[1] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 2